Amendments to the Claims:

Status of Claims:

Claims 1-8, 14, 18-28 are presented for examination.

Claim 28 is added by the present amendment.

Claims 9-13 and 15-17 are canceled by the present amendment.

Claims 1, 14, 18, 23, and 28 are in independent form.

1. (Currently Amended)   A method for distribution of a task, by a host computer, ~~to a device that comprises an address and a processor having an idle state,~~ the method comprising ~~the steps of~~:

formatting ~~the~~ a non-printing task and execution instructions for executing the non-printing task in a packet where the non-printing task is not a task associated with being processed by a printing device;

~~identifying the packet for processing by the processor of the device during a period when the processor is normally in the idle state;~~

transmitting the packet to the printing device and causing the printing device to execute the non-printing task for generation of a result file by ~~the~~ a processor within the printing device in response to the execution instructions; and

receiving the results file by the host computer from the printing device.

2. (Currently Amended)   The method of claim 1 wherein the processor of the printing device runs ~~an alternate~~ a selected personality of a plurality of personalities and the packet is processed by the ~~alternate~~ selected personality.

3. (Currently Amended)   The method of claim 2 wherein the plurality of personalities comprises one or more of a POSTSCRIPT or ~~PCL~~ a printer control language personality.

4. (Currently Amended)   The method of claim 1 wherein the packet is transmitted to the printing device based on an address where the address is a network address.

5. (Original) The method of claim 4 wherein the network address is an Internet protocol address.

6. (Currently Amended) The method of claim [[3]] 4 wherein the network address is an Ethernet address.

7. (Currently Amended) The method of claim 1 and further including the step of the host computer transmitting an executable file along with the packet to the printing device for use by the processor in order to process the non-printing task.

8. (Currently Amended) The method of claim 1 wherein the printing device identifies the non-printing task as an idle state task in response to a port of the printing device over which the packet is received.

9-13. (Canceled)

14. (Currently Amended) A method for distribution of a task, by a host computer, to a printer, where that the printer comprises an operating system that includes a JAVA interpretation process and a processor that executes the JAVA interpretation process and operating system, the printer having an idle state during which printing is not performed, the method comprising the steps of:

identifying the a non-printing task at the host computer where the non-printing task is initially configured to be executed by the host computer and is identified to be distributed to the printer for processing by the printer where the printer functions as a distributed computing device;

formulating the non-printing task into an executable form comprising JAVA code;

wrapping task execution instructions and the JAVA code in a packet;

labeling the packet for processing by the JAVA interpretation process;

transmitting the packet to the printer for generation of results by the JAVA interpretation process in response to the JAVA code and the task execution instructions; and

the host computer receiving the results from the printer.

15-17. (Canceled).

18. (Currently Amended)   A computer system for minimizing processing time for large processing job requests, including a computer ~~in communication with at least one remote peripheral device~~ having a processor, memory, and an operating system, the <u>computer</u> system comprising:

means for parsing tasks from the large processing job request <u>that is originally requested to be executed by the computer</u> ~~for processing by the at least one remote peripheral device~~;

means for generating a <u>non-printing</u> task<u>, from the large processing job request,</u> comprising data and execution instructions <u>configured to allow at least one printing device to execute the non-printing task</u>;

means for wrapping the <u>non-printing</u> task with a functionality label to form a packet;

means for transmitting the packet to the at least one ~~remote peripheral~~ <u>printing</u> device for processing by the at least one ~~remote peripheral~~ <u>printing</u> device to generate task results; and

means for receiving the task results from the at least one ~~remote peripheral~~ <u>printing</u> device.

19. (Currently Amended)   The system of claim 18 and further comprising:

means for receiving the packet at the at least one ~~remote peripheral~~ <u>printing</u> device;

means for determining a necessary functionality for processing the <u>non-printing</u> task from the wrapper label;

means for unwrapping the packet;

means for processing the <u>non-printing</u> task with the necessary functionality, according to the execution instructions, and generating ~~the~~ <u>non-printing</u> task results;

means for capturing the <u>non-printing</u> task results; and

means for addressing the <u>non-printing</u> task results for return to a transmitting computer.

20. (Original) The computer system of claim 19 wherein the necessary functionality is a JAVA Virtual Machine.

21. (Currently Amended) The computer system of claim 19 wherein the at least one ~~remote peripheral~~ printing device is one of a printer, a scanner, gaming systems, ~~and~~ or a personal digital assistant.

22. (Currently Amended) The computer system of claim 19 and further including means for storing the non-printing task in memory of the at least one ~~remote peripheral~~ printing device.

23. (Currently Amended) A computer system for minimizing processing time for large processing job requests, the system comprising:

a computer having a processing unit and memory that stores programming commands that, when read by the processing unit, causes the processing unit to function to: parse a non-printing task from the large processing job request that is awaiting execution by the computer and re-assign the non-printing task for execution by a printing device, and ~~wrap~~ transmit the non-printing task and instructions for processing the non-printing task to the printing device in ~~with a functionality label to form~~ a packet; and

at least one remote ~~peripheral~~ printing device, where a printing device includes ~~having~~ a processing unit and memory that stores programming commands that, when read by the ~~peripheral~~ processing unit, causes the ~~peripheral~~ processing unit of the printing device to function to: receive [[a]] the packet from the computer, wherein the packet includes ~~the functionality label,~~ the non-printing task and the instructions for processing the non-printing task, determine a necessary functionality for processing the non-printing task ~~in response to the wrapper label,~~ unwrapping the packet, and processing the non-printing task with the necessary functionality to generate task results.

24. (Original) The computer system of claim 23 wherein the necessary functionality is a JAVA Virtual Machine.

25. (Currently Amended) The computer system of claim 23 wherein the non-printing task comprises at least one data file.

26. (Currently Amended)   The computer system of claim 23 wherein the <u>non-printing</u> task comprises at least one executable file.

27. (Currently Amended)   The computer system of claim 24 wherein the <u>non-printing</u> task is written by the processor in a code that is interpreted by the JAVA Virtual Machine.

28. (New) A computer-readable medium for providing processor executable instructions for causing a processor to perform a method, the method comprising:

determining that a host computer is to increase processing power for executing a plurality of jobs awaiting execution by the host computer;

selecting a non-print job from the plurality of jobs;

formatting the non-print job to be executable by a printing device;

transmitting the non-print job to the printing device for execution by the printing device; and

receiving an output, by the host computer, from the printing device as a result of executing the non-print job.